BELIEF PROPAGATION DECODER CANCELLING THE EXCHANGE OF UNRELIABLE MESSAGES

Field of the Invention

The present invention relates to a method for decoding at least one codeword comprising N code bits in a decoder. The codeword having been generated in an encoder using a set of M parity equations presentable as a parity check matrix. Further, the present invention relates to a decoder as well as to a mobile terminal and base station employing the decoder. Moreover, the present invention relates to a communication system in which a mobile station and/or a base station according to the invention is/are used.

Related Art

Low-Density Parity-Check Coding

Error correcting codes are widely utilized to obtain reliable communications over noisy channels. Generally speaking, a linear error correcting code C can be described by a parity-check matrix H satisfying Hx = 0 for any codeword $x \in C$. H is an M x N matrix where N is the size of a codeword and M is the number of linear constraints that must be satisfied by each code word. Each row of H therefore represents a linear homogeneous parity-check equation.

Low-density parity-check (LDPC) codes are a particular class of linear error correcting codes characterized by a highly sparse parity check matrix. Typically, in a matrix having a relatively large row length, the entire row may consist of just three ones, the remainder being zeroes. The number of ones in a row *m* is also called the *row weight* ρ_m, while the number of ones in a column *n* is called the *column weight* χ_n. If row weight and column weight are independent of the particular row or column, i.e. $\rho_m=\rho$ and $\chi_n=\chi$ for all *m* and *n*, the code determined by H is called a *regular* code. Otherwise it is called an *irregular* code.

LDPC codes were originally introduced and investigated by Gallager in 1962, (see Gallager, "Low-density parity-check codes", IRE Transactions on Information Theory, Vol. IT-8, pp. 21-28, 1962). A problem of using low density parity check matrices has been to provide a method of decoding, and one of the most significant features of Gallager's work is the introduction of iterative decoding algorithms. He showed that, when applied to sparse parity-check matrices, such algorithms are capable of achieving a significant fraction of the channel capacity in a communication system at relatively low complexity. Furthermore, the number of computations per bit per iteration is independent of the block length N.

Since Gallager's prominent contribution, LDPC codes have been rediscovered and further investigated by Tanner, Wiberg, MacKay and Neal and others. Details of these investigations may be found in Tanner, "A recursive approach to low complexity code", IEEE Transactions on Information Theory, pp. 533-547, 1981, and MacKay et al., "Near Shannon limit performance of Low-Density Parity-Check Codes", IEEE Electronic Letters, vol. 32, pp. 1645-1646, 1996.

Belief Propagation

Known decoders include the maximum likelihood decoder and the maximum a-posteriori decoder. The more widely used of the two, the maximum-likelihood decoder involves finding a most probable information word (where the likelihood is dependent on the channel model). The maximum a-posteriori decoder differs from the maximum likelihood decoding in that it provides an *a posteriori* probability for each symbol. A problem, however, is that the maximum a-posteriori decoder is typically computationally more complex than the maximum-likelihood decoder.

Gallager therefore proposed the iterative decoding scheme referred to above, based on the (later termed) belief propagation method, which approximately converges to the *a posteriori* probability of each symbol.

The method relies on a graph-based representation of codes, where the decoding can be understood as message passing in a factor graph. Belief propagation produces exact probabilities in case of a non-cyclic graph. Unfortunately, the graph associated with an LDPC code is usually cyclic and therefore belief propagation may produce inaccurate probabilities. Nevertheless, Gallager's decoding algorithm gives good empirical performance since, in particular, the end product is the decoding, and so the posterior probabilities need not necessarily be exact.

Sum-Product Algorithm

A brief description of the sum-product iterative algorithm follows. For brevity of exposition, we consider the binary case. The extension to the non-binary case should impose no problem to those skilled in the art, and can be found for example in Davey et al, "Low-Density Parity Check Codes for GF(q)", IEEE Communications Letters, Vol. 2, No. 6, June 1998. Based on the finding that LDPC codes provide a near-Shannon performance when decoded using a probabilistic decoding algorithm, Davey et al. present empirical results of error-correction using the analogous codes over a Galois Field GF(q) for q>2 for different channel models.

More detailed discussion about graphs and the sum-product algorithm, as well as further simplifications for binary variables and parity checks, can be found in Kschischang et al., "Factor Graphs and the Sum-Product Algorithm", IEEE Transactions on Information Theory, Vol. 47, No. 2, February 2001. Kschischang et al. disclose a generic message passing algorithm, the sum-product algorithm, operating in a factor graph. A factor graph is a bipartite graph visualizing the factorization of complicated global functions of many variables.

A simplifying characteristic of the binary case is that since there are only two possible events, the event probabilities can be expressed in terms of a log-likelihood ratio (LLR), which is generally defined by

$$LLR = \ln\frac{p(x=1)}{p(x=0)} = \ln\frac{p(x=1)}{1-p(x=1)} \qquad (1)$$

as the natural logarithm of the ratio of probabilities that x is one of the two possible events.

The algorithm has two alternating components commonly referred to as the horizontal and vertical steps. More specifically, two binary distributions or message matrices, α_{mn} and β_{mn} associated with the non-zero elements h_{mn} of the sparse parity-check matrix H, are iteratively updated. The quantity α_{mn} represents the LLR of the *n*th bit of the transmitted codeword given the information obtained from all the parity equations other than the m^{th} equation. In a similar manner, β_{mn} represents the LLR for the satisfaction of the m^{th} parity-check equation given the LLR of the n^{th} bit and all the other bits are statistically independent, with associated distributions $\beta_{m1}...\beta_{mN}$. Assuming that the codewords are used with equal probability on an arbitrary binary-input continuous-output channel, the sum-product algorithm can be described as follows.

Before summarizing the algorithm, some terms used in the following sections are defined first. The terms are described within the error decoding context. Fig. 1 shows a sample parity-check matrix H, together with a graph representation of the matrix which may be used to confer the following definitions.

A Check Node represents one parity check bit. A parity check bit is generally computed from one or more information bits such that a given equation is solved. In binary logic, an equation might be represented by the modulo-2 sum of information bits, where the results should be equal to the parity check bit.

A Variable Node represents one coded bit. In communication decoding a coded bit is usually equivalent to the received value of a code bit, or for example a log-likelihood ratio. A Variable Node is connected to one or more Check Nodes, which represents the relation of that variable to the respective parity check bit. In other words, a variable participates in all the equations of connected Check Nodes.

An edge between a Variable Node and a Check Node is used to represent a relation between these. On the Variable Node side the edges to Check Nodes give information which check equations involve that variable. On the Check Node side the edges to Variable Nodes give information which variables are involved in that particular parity check equation.

In the decoding process a message is passed along the edges from Variable Node to Check Node, and also from Check Node to Variable Node. The message typically contains some sort of belief or probability information (hence "belief propagation" is widely used in literature).

The Sum-Product algorithm may be defined by the following steps:

Initialization

λ_n represents the a priori LLR of the *n*th bit of the transmitted codeword. This LLR can be calculated from the received vector corresponding to the transmitted codeword and the channel model. For each non-zero entry h_{mn} of the parity-check matrix H, set:

$$\alpha_{mn} = \lambda_n \qquad (2)$$

Horizontal step

For each non-zero entry h_{mn} of the matrix H, β_{mn} is computed using the equation:

$$\beta_{mn} = 2 \cdot \tanh^{-1} \prod_{\substack{l=1..N \\ l \neq n}} \tanh \frac{\alpha_{ml}}{2} \tag{3}$$

where l runs over the non-zero bit positions of the m^{th} parity equation (i.e. $h_{ml} \neq 0$), excluding the n^{th} position.

Fig. 2 shows a graph representation of an exemplary parity check matrix

$$H = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

The horizontal step can be visualized according to Fig. 2 as follows. Whenever a β_{mn} is to be calculated for a given edge *mn* connecting Check Node *m* to Variable Node *n*, all α values on edges are involved which are connected to Check Node *m* except the α value which belongs to edge *mn*, i.e. α_{mn}.

Vertical Step

For each non-zero entry h_{mn} of the matrix H, update α_{mn} in the following manner:

$$\alpha_{mn} = \lambda_n + \sum_{\substack{j=1..M \\ j \neq m}} \beta_{jn} \tag{4}$$

where j runs over the parity equations for which $h_{jn} \neq 0$.

Decoding

A single iteration typically consists of an application of the horizontal step and then of an application of the vertical step (except for the first iteration where additionally α_{mn} is initialized based on the symbols received from the channel). At the end of any iteration (including the initialization step) one can also update the soft-output decision (i.e. the estimated posteriori LLR ξ_n) of each bit:

$$\xi_n = \lambda_n + \sum_{j=1..M} \beta_{jn} \tag{5}$$

where j runs over the parity equations for which $h_{jn} \neq 0$. Using equation (5) the value of the *k*th transmitted bit can be estimated as

$$\hat{x}_k = \begin{cases} 1 & if\xi_n \geq 0 \\ 0 & if\xi_n < 0 \end{cases} = \begin{cases} 1 & if\ \lambda_n + \sum_{j=1..M}\beta_{jn} \geq 0 \\ 0 & if\ \lambda_n + \sum_{j=1..M}\beta_{jn} < 0 \end{cases} \qquad (6)$$

In this manner a tentative bit-by-bit decoding is performed and a vector *x* is obtained. If Hx=0, i.e. if ξ is a codeword, the decoding algorithm halts declaring x as the output. Otherwise, the iterative process continues by returning to the horizontal step above. The decoding procedure terminates by declaring a decoding-failure, or if some maximum predetermined number of iterations (e.g. 100) occurs with no successful decoding.

Min-Sum Algorithm

To simplify the calculations involved, equation (3) in the horizontal step may be approximated and substituted by

$$\beta_{mn} = \min_{\substack{l=1..N \\ l\neq n}}(|\alpha_{ml}|) \cdot \prod_{\substack{l=1..N \\ l\neq n}} \mathrm{sgn}(\alpha_{ml}) \qquad (7)$$

This approximation will degrade the performance of the decoding however.

WO 02/37731 A2 provides an implementation of the sum-product algorithm which uses different sets of equations to achieve the horizontal step, vertical step, and decoding. More particularly, the reference employs a likelihood difference as also defined by Kschischang et al. to perform the calculations of the horizontal and vertical step.

As can be seen from the equations above for the horizontal step, vertical step, and decoding, the information from numerous values is involved which is ultimately derived from the received vector corresponding to the transmitted codeword. In a noisy channel environment, chances are high that several received values (bits) carry wrong information, which implies that wrong information can be inferred from these values and propagate through the decoding iterations.

Summary of the Invention

It is therefore the object of the present invention to reduce the influence of wrong information in an iterative decoding method.

The object of the present invention is solved by the subject matter of the independent claims. Preferred embodiments of the present invention are subject matter to the dependent claims.

In the following, the terminology used in the introduction and by Kschischang et al. has been retained to facilitate the comprehension of the present invention. According to the present invention a message matrix may be a rectangular matrix which holds generally real values representing the messages about a certain coded symbol. The message itself may be a knowledge-related, belief-related, or probability-related measure. Generally the position within the matrix gives information about the concerned coded symbol, as well as about how the message is obtained. In a preferred embodiment the row index denotes the parity check equation which is relevant for the message, while the column gives information about the coded symbol for which the message is valid.

According to an embodiment of the present invention a method for decoding at least one codeword x comprising N code bits in a decoder is provided. The codeword x may have been generated in an encoder using a set of M parity equations presentable as a parity check matrix H. In other words, the decoding is performed as message passing on a graph representation of the used code, wherein the graph representation is based on a parity check matrix.

According to the method, for each non-zero entry in a parity check matrix H, the elements α_{mn} of a first message matrix may be initialized with data λ_n obtained from a demodulator. Next, for each non-zero entry in the parity check matrix H, the elements β_{mn} of a second message matrix may be determined based on elements α_{mn} of said first message matrix, and a decoded codeword $\hat{x}$ may be reconstructed based on the data λ_n obtained from the demodulator and the elements β_{mn} of the second message matrix. According to this embodiment, only a subset of message matrix elements α_{mn} from said first message matrix may be used for determining a message matrix element β_{mn} of said second message matrix, wherein the message matrix elements of said subset fulfill a reliability criterion.

According to the method, only reliable message matrix elements of the first message matrix, e.g. the elements α_{mn}, may be used to determine the message matrix elements β_{mn} of a second matrix. Taking the example of the Sum-Product algorithm as outlined above, in any of the equations of the horizontal step and/or vertical step, not all information may be processed. Instead some of the values, i.e. message matrix elements may be excluded from the calculations. Which values are excluded may be determined according to criteria as outlined further down below. Generally a value may be excluded if it is unreliable such that the inclusion of the term would produce worse results than its omission from the respective equation.

Further, the message matrix elements α_{mn} of the first message matrix may be updated based on a subset of message matrix elements β_{mn} of said second message matrix, wherein the message matrix elements of said subset fulfill a reliability criterion. This may for example be applicable in case it has been decided to perform several iterations before trying to decode a codeword. Hence, when considering the example of the Sum-Product algorithm again, the calculations of the α_{mn} values in the vertical step may be performed on a reduced set of elements β_{mn}.

In the embodiment above only the message matrix elements β_{mn} of the second message matrix may be updated. According to a further embodiment it is also possible to update only the message matrix elements α_{mn} based on a reduced set, i.e. a subset of message matrix elements β_{mn}.

Therefore, the present invention further provides a method for decoding at least one codeword x comprising N code bits in a decoder wherein the codeword x may have been generated in an encoder using a set of M parity equations presentable as a parity check matrix H. According to this embodiment, for each non-zero entry in a parity check matrix H, the elements α_{mn} of a first message matrix may be initialized with data λ_n obtained from a demodulator. Next, for each non-zero entry in the parity check matrix H, the elements β_{mn} of a second message matrix may be determined based on elements α_{mn} of said first message matrix, and a decoded codeword $\hat{x}$ may be reconstructed based on the data λ_n obtained from the demodulator and the elements β_{mn} of the second message matrix. Further, for each non-zero entry in the parity check matrix H, the message matrix elements α_{mn} of the first message matrix may be updated based on message matrix elements β_{mn} of the second message matrix, wherein a subset of message matrix elements β_{mn} of said second message matrix may be used to update a message matrix element α_{mn} of said first message matrix, wherein the message matrix elements of said subset fulfill a reliability criterion.

According to a further embodiment of the present invention, only a subset of message matrix elements α_{mn} of said first message matrix is used for determining a message matrix element β_{mn} of said second message matrix, wherein the message matrix elements of said subset fulfill a reliability criterion.

In another embodiment, the elements of the first and/or the second message matrix may be recalculated in several iterations before the decoding method tries to decode the received code bits for the first time. Hence, the method proposed may further comprise the step of performing iterations by repeating the determination of the message matrix elements β_{mn} of the second message matrix.

In a further embodiment, it may be repeatedly tried to decode the code bits received. Thus, the iterations may further comprise reconstructing a decoded codeword $\hat{x}$ based on the data λ_n obtained from the demodulator and the elements β_{mn} of the second message matrix in each iteration.

In order to determine, whether the reconstructed codeword is a valid codeword, it may be checked whether the parity check equations of the parity-check matrix H are satisfied by the reconstructed decoded codeword $\hat{x}$.

The iteration through the different processing steps as outlined above may be stopped upon reaching on predetermined number of iterations or in case the decoded codeword $\hat{x}$ satisfies the parity check equations.

In another embodiment of the present invention, the data obtained from a demodulator may comprise parameters representing one of a likelihood ratio, a likelihood difference or a probability for each of the N code bits of a codeword x. Thus, different measures or parameters may be used in the proposed method to calculate the different elements of the first and second message matrix.

Further, the data obtained from the demodulator may further comprise indications which of the parameters provided fulfill the reliability criterion.

According to a further embodiment of the present application, the subset of message matrix elements β_{mn} of said second message matrix used to determine the message matrix elements α_{mn} of said first message matrix may be updated upon determining new message matrix elements β_{mn} in an iteration step. This allows adapting the decoding method to a possibly redefined reliability criterion in each iteration step.

When updating the subset, the updated subset may therefore only comprise message matrix elements β_{mn} of the second message matrix fulfilling a reliability criterion, e.g. a criterion updated during the different possible iterations in the decoding process.

In another embodiment of the present invention also the subset of message matrix elements α_{mn} of said first message matrix used to determine the message matrix elements β_{mn} of said second message matrix may be updated upon determining new message matrix elements α_{mn} in an iteration step.

When updating the subset, the updated subset may comprise only message matrix elements α_{mn} of the first message matrix fulfilling the reliability criterion.

According to a further embodiment, the reliability criterion may be based on at least one of channel estimations of a radio channel via which the codeword x has been received, the absolute values of the elements of the first and/or second message matrix, the absolute values of the data provided by the demodulator, the number of the iteration in the decoding which have already been processed, the maximum number of iterations to be performed in the decoding process, and a random process.

It may be determined that the reliability criterion is not fulfilled by a message matrix element of the first or the second message matrix, if the signal to noise ratio for the element and/or the absolute value of the element is below a predetermined threshold value.

According to this embodiment, it may be for example possible, to use the channel estimations of the radio channel as a basis for deciding which of the initialized message matrix elements α_{mn} of the first message matrix are added to the subset used to determined the message matrix elements β_{mn} of the second message matrix. Next, when updating either the first or second message matrix or both, the absolute value of individual message matrix elements may build the basis for the reliability criterion to define the subset for updating. Also the number of iterations already performed may influence the reliability criterion.

In a further embodiment of the present invention, the error correcting code used is a low-density parity-check (LDPC) code.

Another embodiment of the present invention is related to a decoder. The decoder may be used for decoding at least one codeword x and may comprise processing means for initializing the elements α_{mn} of a first message matrix for each non-zero entry in a parity check matrix H with data λ_n obtained from a demodulator, for determining the elements β_{mn} of a second message matrix for each non-zero entry in the parity check matrix H, based on elements α_{mn} of said first message matrix, and for reconstructing a decoded codeword $\hat{x}$ based on the data λ_n obtained from the demodulator and the elements β_{mn} of the second message matrix.

Moreover, the processing means may be further adapted to use a subset of message matrix elements α_{mn} from said first message matrix for determining a message matrix element β_{mn} of said second message matrix, wherein the message matrix elements of said subset fulfill a reliability criterion.

According to an alternative embodiment of the present invention, a decoder for decoding at least one codeword x may comprise processing means for initializing the elements α_{mn} of a first message matrix for each non-zero entry in a parity check matrix H with data λ_n obtained from a demodulator, for determining the elements β_{mn} of a second message matrix for each non-zero entry in the parity check matrix H, based on elements α_{mn} of said first message matrix, and for reconstructing a decoded codeword $\hat{x}$ based on the data λ_n obtained from the demodulator and the elements β_{mn} of the second message matrix, and for updating the message matrix elements α_{mn} of the first message matrix for each non-zero entry in the parity check matrix H based on message matrix elements β_{mn} of the second message matrix.

Further the processing means may be adapted to use a subset of message matrix elements β_{mn} of said second message matrix to update a message matrix element α_{mn}

of said first message matrix, wherein the message matrix elements of said subset fulfill a reliability criterion.

The two alternative implementations of the decoder mentioned above may be further adapted to perform the decoding methods described above.

In other embodiments, the present invention provides a mobile terminal and a base station in a mobile communication system, both comprising receiving means for receiving at least one codeword x, demodulation means for demodulating the at least one received codeword x and for delivering data to a decoder, and the decoder according to one of the various embodiments described above.

The mobile terminal and/or the base station may further comprise coding means for encoding data in at least one codeword x, and transmission means for transmitting the at least one codeword x, and in that at least one transmitted codeword x is suitable for decoding according to the different decoding methods described above.

Moreover the present invention further provides a mobile communication system comprising at least one base station and at least one mobile terminal.

Brief Description of the Figures

In the following the present invention is described in more detail in reference to the attached figures and drawings. Similar or corresponding details in the figures are marked with the same reference numerals.

Fig. 1 shows a sample parity-check matrix H, together with a graph representation of the matrix,

Fig. 2 shows the graph representation of Fig. 1 together with the messages α_{mn} and β_{mn} which are passed along the edges in belief propagation algorithms,

Fig. 3 shows a flow chart of an exemplary decoding process according to an embodiment of the present invention,

Fig. 4 shows a flow chart of another exemplary decoding process according to an embodiment of the present invention,

Fig. 5 shows a transmitter and a receiver unit according to an embodiment of the present invention,

Fig. 6 shows a mobile terminal according to an embodiment of the present invention comprising the transmitter and the receiver shown in Fig. 5,

Fig. 7 shows a base station according to an embodiment of the present invention comprising the transmitter and the receiver shown in Fig. 5, and

Fig. 8 shows an architectural overview of a communication system according to an embodiment of the present invention comprising a mobile terminal shown in Fig. 6 and a base station (Node B) shown in Fig. 7.

Detailed Description

In the following description of the different embodiments of the present invention the expression "$x \in A \setminus B$" denotes "x is element of set A without set B", which is equivalent to "x is element of set A but not element of set B". Further, the following paragraphs will outline the ideas underlying the present invention by way of example considering LDPC decoding. However, it should be noted that the principles underlying the present invention may also be applicable to other codes.

As outlined before, mathematical equations may be solved in the horizontal step, vertical step, and decision step. At least the horizontal step and vertical step equations are computed in each iteration, such that it would be formally correct to refer to e.g. $\alpha_{mn}^{(i)}$ and $\beta_{mn}^{(i)}$, where *i* represents the iteration number, and the initialization step for α_{mn} may be interpreted as iteration number zero, i.e. $\alpha_{mn}^{(0)}$. However for sake of simplicity and brevity an iteration superscript has been omitted from the formulas.

Generally, it should be noted that α values are necessary to compute β values in the horizontal step. Likewise it should be noted that β values are necessary to compute α values in the vertical step as well as to compute ξ values in the decoding process.

A new β_{mn} value, i.e. a new message matrix element, is computed from α_{ml} values where l takes all values from 1 to N where the parity-check matrix entry h_{ml} is not zero, except *n*. This can be interpreted as l being element of a set L_{mn}.

$$L_{mn} = \{l \in [1, n-1] \cup [\mathrm{n}+1, \mathrm{N}] | h_{ml} \neq 0\} \qquad (8)$$

L_{mn} represents the set of values $l \neq n$ between 1 and N for which $h_{ml} \neq 0$.

Using equation (8) the horizontal step may be reformulated:

$$\beta_{mn} = 2 \cdot \tanh^{-1} \prod_{l \in L_{mn}} \tanh \frac{\alpha_{ml}}{2} \tag{9}$$

or alternatively according to equation (7) approximated and simplified to

$$\beta_{mn} = \min_{l \in L_{mn}} \left(|\alpha_{ml}| \right) \cdot \prod_{l \in L_{mn}} \operatorname{sgn}(\alpha_{ml}) \tag{10}$$

Similarly there may exist a vertical step set J_{mn} which can be defined as follows:

$$J_{mn} = \left\{ j \in [1, m-1] \cup [\mathrm{m}+1, \mathrm{M}] \,\middle|\, h_{jn} \neq 0 \right\} \tag{11}$$

J_{mn} may be interpreted as a set of values $j \neq m$ between 1 and M for which $h_{jn} \neq 0$.

With equation (11) the vertical step can be rewritten as

$$\alpha_{mn} = \lambda_n + \sum_{j \in J_{mn}} \beta_{jn} \tag{12}$$

According to the present invention, exclusion sets Γ_{mn} and Ω_{mn} may be defined for the horizontal and vertical steps. The exclusion set may comprise message matrix elements which are not considered when determining or updating the message matrix elements α_{mn} of the vertical step and/or the message matrix coefficients β_{mn} of the horizontal step.

A new horizontal step may be defined as:

$$\beta_{mn} = 2 \cdot \tanh^{-1} \prod_{l \in L_{mn} \setminus \Gamma_{mn}} \tanh \frac{\alpha_{ml}}{2} \tag{13}$$

or alternatively approximated and simplified to

$$\beta_{mn} = \min_{l \in L_{mn} \setminus \Gamma_{mn}} \left(|\alpha_{ml}| \right) \cdot \prod_{l \in L_{mn} \setminus \Gamma_{mn}} \operatorname{sgn}(\alpha_{ml}) \tag{14}$$

A new vertical step may be chosen as follows:

$$\alpha_{mn} = \lambda_n + \sum_{j \in J_{mn} \setminus \Omega_{mn}} \beta_{jn} \tag{15}$$

If both sets Γ_{mn} and Ω_{mn} are empty, prior art behavior is replicated. If Γ_{mn} contains the same elements as L_{mn} and/or Ω_{mn} contains the same elements as J_{mn} for all possible values of m and n, the result would be no message propagation at all, which is equivalent to no decoding.

The situation where Γ_{mn} contains the same elements as L_{mn} may occur for arbitrary values of m and n. In such a case, the respective β_{mn} values in the horizontal iteration step may be set to zero.

In case Γ_{mn} contains the same elements as L_{mn} for all values of m and n, all β_{mn} values may be set to zero. Consequently the next vertical step may be interpreted as a re-initialization of the α_{mn} (see equation (15)), i.e. setting all $\alpha_{mn} = \lambda_n$. Thus, in such a case the decoded codeword $\hat{x}$ may correspond to a decision based directly on λ_n (see equation (6)). From a technical point of view this may be regarded equivalent to not decoding the inputted data but to simply pass them to the next processing entity. It should be noted that in case Γ_{mn} contains the same elements as L_{mn} for all m and n, according to equation (15) the calculation of α_{mn} becomes independent from the exclusion set Ω_{mn}, as any β_{mn} element involved would have been set to zero. In this case also no effective message propagation among Nodes is performed.

Conversely, the initialization step may be viewed as an instance of a single new vertical step where the exclusion set Ω_{mn} contains the same elements as J_{mn}. Alternatively the initialization step may also be viewed as an instance of a single vertical step where all β_{mn} values are equal to zero.

Generally the exclusion sets are depending on the reliability criteria, and therefore can depend on parameters such as the row index m for which an equation is solved, the column index n for which an equation is solved and/or the iteration i of the whole decoding algorithm. Further, the exclusion sets may be updated during the decoding process in dependence on decision criteria further outlined below.

As an example, in the horizontal step the message matrix element β_{23}^{i} in iteration step i (see Fig. 2) may be determined by calculating

$$\beta_{23}^{i} = \min_{l\in L_{23}^{i}\backslash\Gamma_{23}^{i}=\{\}}\left(\left|\alpha_{2l}^{i-1}\right|\right)\cdot\prod_{l\in L_{23}^{i}\backslash\Gamma_{23}^{i}=\{\}}\mathrm{sgn}\left(\alpha_{2l}^{i-1}\right)=\min\left(\left|\alpha_{22}^{i-1}\right|,\left|\alpha_{26}^{i-1}\right|\right)\cdot\mathrm{sgn}(\alpha_{22}^{i-1})\cdot\mathrm{sgn}(\alpha_{26}^{i-1})$$

assuming that Γ_{23}^{i} is empty and $L_{23}^{i}=\{2,6\}$. In case it is determined that a newly calculated α_{22}^{i} does not fulfill the reliability criterion, e.g. $\left|\alpha_{22}^{i}\right|<\alpha_{22}^{threshold}$, the exclusion set Γ_{23}^{i} may be updated to $\Gamma_{23}^{i+1}=\{2\}$ in the next iteration *i+1*, such that the calculation of the new β_{23}^{i+1} would only consider α_{26}^{i}, i.e.

$$\beta_{23}^{i+1} = \min_{l\in L_{23}^{i+1}=\{2,6\}\backslash\Gamma_{23}^{i+1}=\{2\}}\left(\left|\alpha_{2l}^{i}\right|\right)\cdot\prod_{l\in L_{23}^{i+1}=\{2,6\}\backslash\Gamma_{23}^{i+1}=\{2\}}\mathrm{sgn}\left(\alpha_{2l}^{i}\right)=\left|\alpha_{26}^{i}\right|\cdot\mathrm{sgn}(\alpha_{26}^{i})=\alpha_{26}^{i}$$

As can be seen in the example above, if the value α_{22}^{i} were unreliable, the resulting β_{23}^{i+1} value could have been forged by the unreliable message matrix element α_{22}^{i}. E.g. assuming that α_{26}^{i} correctly indicates a positive sign but α_{22}^{i} would indicate a negative sign, the resulting β_{23}^{i+1} would have been assumed to be negative in its sign as well. Thus, when determining a decoded codeword $\hat{x}$ according to equation (6) a wrong codeword $\hat{x}$ may be obtained if one or several of the β_{mn}^{i+1} elements are corrupted in a similar manner.

Exclusion sets may be defined in order to exclude data from the equations (or decoding process) which are assumed to be wrong, or which are highly likely to be wrong. Another possible definition criterion may be whether a message matrix element or value may provide a contribution to the decision process. E.g. if an LLR value is close to 0, this means that both events represented by this LLR are almost equally likely, such that no information (or only little to be correct) may be obtained from this value with regard to the correct decoding of a codeword.

If such data is included, the produced output is likely to be wrong as well. Therefore the present invention proposes to neglect such values from the equations as disclosed above.

As mentioned above, the exclusion sets for the new horizontal and vertical steps should be defined such that unreliable messages are excluded from the calculations. It should be obvious to those skilled in the art that the exclusion sets may be defined independently from each other. In other words an element of exclusion set Γ_{mn} does not necessarily be element of exclusion set Ω_{mn}.

Similarly the exclusion sets may be set independently in decoding iterations. Those skilled in the art will recognize that with increasing number of iterations, the overall reliability of messages passed may be increasing for reasonably good transmission conditions. Therefore with increasing number of iterations the number of elements of the exclusion sets may be reduced, such that at late stages of decoding the exclusion sets may be empty. It should be noted that the exclusion sets may depend both on the number of iterations processed so far, as well as on the maximum number of decoding iterations, which may be a parameter given by the communication system.

A list of possible criteria which may be used isolated or in combination for determining the exclusion sets is provided in the following. One parameter on which the reliability criterion deciding whether a message matrix element is comprised in exclusion set or not may be results from channel estimations. The channel conditions for each bit of a received codeword do not necessarily have to be similar, e.g. in case of employing OFDM (Orthogonal Frequency Division Multiplex). In such a system the different bits of a codeword may be transmitted on different subcarriers of a subcarrier set (channel). Hence, different channel conditions on the subcarriers may exist. As a measure for the channel conditions the SNR (Signal to Noise Ratio) or related quantities may be employed.

Another parameter on which a reliability criterion may be based on is the absolute values of the measures involved in the horizontal or vertical step, i.e. the log-likelihood ratio, the likelihood ratio, a likelihood difference, a log-likelihood difference, and/or another probability measure. As outlined above for LLR values, an absolute LLR value close to 0 may not provide any significant information for the decoding process.

As outlined above as well the iteration number of the decoding process may also provide a basis for a reliability criterion to determine the exclusion sets Γ_{mn} and Ω_{mn}. Another alternative basis may be a random process deciding which of the available message matrix elements should be included in the exclusion set, i.e. which of the message matrix parameters are excluded from the calculations in the horizontal and vertical steps.

It is further noted that the reconstruction of a codeword $\hat{x}$ may be performed after each horizontal step, such that the vertical step is only executed if the decoding procedure should be continued. Those skilled in the art will recognize that this is an implementational detail of the algorithm which does not affect the computational results.

As will be shown in the following section the ideas underlying the present invention may also be applied to decoding methods in which the calculations performed in the horizontal and/or vertical steps are based on a (log-)likelihood difference. Such a decoding method is for example known from WO 02/37731 A2. The exemplary system defined therein defines the horizontal step as:

$$r_{ik}(0)=\frac{1}{2}\left(1+\prod_{l\neq k}\delta q_{il}\right) \qquad (16)$$

$$r_{ik}(1)=\frac{1}{2}\left(1-\prod_{l\neq k}\delta q_{il}\right) \qquad (17)$$

and the vertical step as:

$$q_{ik}(0)=\alpha\cdot p_k(0)\prod_{j\neq i}r_{jk}(0) \qquad (18)$$

$$q_{ik}(1)=\alpha\cdot p_k(1)\prod_{j\neq i}r_{jk}(1) \qquad (19)$$

According to the present invention, a new horizontal and vertical step may be defined by:

$$r_{ik}(0)=\frac{1}{2}\left(1+\prod_{\substack{l\neq k\\ l\notin\Gamma_{ik}}}\delta q_{il}\right) \qquad (20)$$

$$r_{ik}(1)=\frac{1}{2}\left(1-\prod_{\substack{l\neq k\\ l\notin\Gamma_{ik}}}\delta q_{il}\right) \qquad (21)$$

for the horizontal step and by

$$q_{ik}(0)=\alpha\cdot p_k(0)\prod_{\substack{j\neq i\\ j\notin\Omega_{ik}}}r_{jk}(0) \qquad (22)$$

$$q_{ik}(1)=\alpha\cdot p_k(1)\prod_{\substack{j\neq i\\ j\notin\Omega_{ik}}}r_{jk}(1) \qquad (23)$$

for the vertical step.

The notation used in this embodiment of the present invention above is similar to that of WO 02/37731 A2, except for the exclusion sets Γ and Ω, which are used as described in the present invention. δq_{ik} refers to the difference between the probabilities that the ith element in the kth row is a "1" and a "0", i.e. $\delta q_{ik} = q_{ik}(1) - q_{ik}(0)$.

Further, $p_k(1)$ is the prior probability that the kth bit of the received codeword is a "1" (i.e. k=1..N). Accordingly, $p_k(0)$ is the prior probability that the kth bit of the received codeword is a "0". Hence, in the example algorithm given above, the two probabilities correspond to the initialization data λ_n, or in more detail $\lambda_n = \ln\frac{p_n(1)}{p_n(0)}$. In equations 16 through 23, α is a scaling factor chosen such that $q_{ik}(1) + q_{ik}(0) = 1$.

To decide whether a bit in the codeword is a "1" or a "0" the equation

$$\hat{x} = \begin{cases} 1 & for\ q_k(1) \geq q_k(0) \\ 0 & for\ q_k(1) < q_k(0) \end{cases} \qquad (24)$$

may be used where

$$q_k(0) = \alpha \cdot p_k(0) \cdot \prod_{j=1..M} r_{jk}(0) \qquad (25)$$

and

$$q_k(1) = \alpha \cdot p_k(1) \cdot \prod_{j=1..M} r_{jk}(1) \qquad (26).$$

After having discussed the underlying ideas of the present invention in detail, the following section will describe preferred embodiments of decoding processes of the present invention.

Fig. 3 shows a flow chart of an exemplary decoding process according to an embodiment of the present invention. In a fist step 301, a counter *i* for counting the iteration of the decoding process may be initialized. Next, the message matrix elements α_{mn} may be initialized 302 e.g. by using the receiving means estimate of each codeword bit λ_n in form of an LLR value and the exclusion set Γ_{mn} may further be initialized based on parameters, e.g. a SNR value, indicating which of the codeword bits λ_n fulfill a reliability criterion as defined in the previous sections. Optionally, in step 302 the exclusion set Ω_{mn} may further be initialized e.g. by defining an empty set or by including the elements of Γ_{mn}.

Taking the example of using likelihood differences as a probability measure, as described above, the initialization may be performed similarly. Instead of using the input data λ_n, the receiving section may provide the probabilities $p_k(0)$ and $p_k(1)$ which may be used to initialize the quantities $q_{ik}(0)$ and $q_{ik}(1)$. In this example decoding process $q_{ik}(0)$ and $q_{ik}(1)$ may be considered as corresponding to the usage of the message matrix elements α_{mn} during initialization.

Upon having initialized the message matrix elements α_{mn} and the exclusion set Γ_{mn}, the message matrix elements β_{mn} of the horizontal step may be calculated 303 according to one of the equations (13) or (14) while taking into account the exclusion set Γ_{mn}.

When using likelihood differences as a probability measure, step 303 may correspond to determining the measures $r_{ik}(0)$ and $r_{ik}(1)$ as suggested by equations (20) and (21), under consideration of the exclusion set Γ_{mn}.

According to this embodiment of the present invention, the decoder may next try to reconstruct the received codeword $\hat{x}$, e.g. by using equation (6). Accordingly, equation (24) may be used when employing likelihood differences.

Next, it may be checked whether the reconstructed codeword $\hat{x}$ satisfies 305 the parity equations, e.g. by checking whether $H\hat{x}=0$. If all parity check equations are satisfied, the decoding process may be stopped 306 and the successfully reconstructed codeword $\hat{x}$ may be provided to a further processing entity in the decoder. If at least one of the parity check equations is not satisfied, it may be determined if the maximum number of iterations as been reached 307. If so, the decoding process may be stopped 308 and a decoding error may be indicated.

If the number of iterations has not reached i_{max} the exclusion set Ω_{mn} may be determined 309 based on the calculated message matrix elements β_{mn} and/or the respective reliability criterion for each/the message matrix element/s.

Using the determined exclusion set Ω_{mn} new message matrix elements α_{mn} may be calculated 310 based on equation (15), which corresponds to the vertical step. When employing likelihood differences, equations (20) and (21) may be used to calculate the new $r_{ik}(0)$ and $r_{ik}(1)$ values.

Upon having determined the new message matrix elements α_{mn} (or the new $r_{ik}(0)$ and $r_{ik}(1)$ values) the exclusion set Γ_{mn} may be updated 311 accordingly. Upon incrementing the iteration counter 312, the processing may continue by calculating new β_{mn} values as outlined above (see step 303).

Next, a further embodiment of a decoding process of the present invention will be outlined in reference to Fig. 4. Fig. 4 shows a flow chart of another exemplary decoding process according to an embodiment of the present invention. The initialization of the decoder as shown in steps 401 and 402 correspond to those known from Fig. 3 (see steps 301, 302). Also the calculation of the β_{mn} in step 403 may be identical to step 303 of Fig. 3.

In contrast to the decoding process known from Fig. 3, the decoder does not try to reconstruct a decoded codeword $\hat{x}$ in each iteration step, but a number of iterations are performed (see steps 403 to 408) first before trying to reconstruct codeword $\hat{x}$ 409.

Upon having calculated the β_{mn} in step 403, it may be determined if the maximum number of iterations as been reached 404. If so, the decoding process may jump to step 409.

If the number of iterations has not reached i_{max} the exclusion set Ω_{mn} may be determined 405 based on the calculated message matrix elements β_{mn} and/or the respective reliability criterion for each/the message matrix element/s.

Using the determined exclusion set Ω_{mn} new message matrix elements α_{mn} may be calculated 406 based on equation (15), which corresponds to the vertical step. When employing likelihood differences, equations (20) and (21) may be used to calculate the new $r_{ik}(0)$ and $r_{ik}(1)$ values.

Upon having determined the new message matrix elements α_{mn} (or the new $r_{ik}(0)$ and $r_{ik}(1)$ values) the exclusion set Γ_{mn} may be updated 407 accordingly. Upon incrementing the iteration counter 408, the processing may continue by calculating new β_{mn} values as outlined above (see step 403).

When exiting the iteration loop, i.e. when the iteration number *i* equals i_{max}, the decoder may reconstruct a codeword $\hat{x}$ 409. Further, it may be checked 410 if the reconstructed codeword $\hat{x}$ fulfills the parity check equations represented by parity check matrix H.

If all parity check equations are satisfied, the decoding process may be stopped 411 and the successfully reconstructed codeword $\hat{x}$ may be provided to a further processing entity in the decoder.

If at least one of the parity check equations is not satisfied, the decoding process may be stopped 308 and a decoding error may be indicated. Alternatively, the counter *i* may be reset and the maximum number of iterations may be reconfigured i_{max} and further iterations according to steps 403 to 408 may be performed, before trying to reconstruct a codeword $\hat{x}$ again.

Next, Fig. 5 will be discussed in more detail. Fig. 5 shows a transmitter and a receiver unit according to an embodiment of the present invention. The transmitter 501 comprises an encoder 502 and a transmission means 503. The transmission means may comprise a modulator for modulating the signals encoded by encoder 502. As indicated by the dotted arrow, the encoder 502 is capable of encoding input data into codeword suitable for decoding according to the various embodiments of the decoding process. The modulated data may be transmitted by the transmission means 503 using an antenna as indicated.

The receiver 504 receiving the encoded signals may comprise a receiving means 506, which may comprise a demodulator for demodulating the received signals. Upon extracting the λ_n values and parameters in the receiving means 506, these data may be provided to a decoder 505, which will consider the data to initialize the decoding process as outlined above.

The decoder 505 may comprise a processing means 507, adapted to decode the received data according to the methods described to produce reconstructed codewords.

Fig. 6 and 7 show a mobile terminal 601 and a base station (Node B) 701 according to different embodiments of the present invention, respectively. The mobile terminal 601 and the base station may each include a transmitter 501 and a receiver 504 as shown in Fig. 5 to perform communications.

Fig. 8 shows an architectural overview of a communication system according to an embodiment of the present invention comprising a mobile terminal 601 shown in Fig. 6 and a base station (Node B) 701 shown in Fig. 7.

The overview depicts a UMTS network 801, which comprises a core network (CN) 803 and the UMTS terrestrial radio access network (UTRAN) 802. The mobile terminal 601 may be connected to the UTRAN 802 via a wireless link to a Node B 701. The base stations in the UTRAN 802 may be further connected to a radio network controller (RNC) 804. The CN 803 may comprise a (Gateway) Mobile Switching Center (MSC) for connecting the CN 803 to a Public Switched Telephone Network (PSTN). The Home Location Register (HLR) and the Visitor Location Register (VLR) may be used to store user related information. Further, the core network may also provide connection to an Internet Protocol-based (IP-based) network through the Serving GPRS Support Node (SGSN) and the Gateway GPRS Support Node (GGSN).